# Matthew Weber

CONTACT INFORMATION

2301 Grand AVE S
Minneapolis, MN 55405

*Phone:* (612) 787-8902
*E-mail:* matt@badecho.com

## PROFILE

Dedicated individual with 12+ years experience in a lead developer role who is able to provide excellent software design and execution of its implementation. By leading others through teaching and example, total contribution to the company is amplified by ensuring everyone else's contribution is where it should be. Track record demonstrates an extremely high rate of output while working.

## ACCOMPLISHMENTS

- Implemented a large number of features, ranging from user interface to systems level, in a new software product responsible for performing stress tests through its operation of large testing rigs that simulate wear and tear on vehicles using recorded road data.
- Solely responsible for the design and implementation of the event handling subsystem of an automobile test suite, which would execute user defined actions in the response to real world events such as unsafe operating conditions, etc.
- Cleaned up code, modernized, and implemented new features in the user interface of a very large software system used by audiologists to configure hearing aid devices during a patient fitting session.
- Worked on porting code from a legacy technology stack to a next generation system responsible for communicating with hearing aid device hardware and configuring the multitude of features made available by said hardware products.
- Designed and implemented an innovative software system and rapid development platform, which served as the foundation for new software released as the next version of the company's flagship product. This platform consisted of frameworks purposed for the rapid creation of both client and server products, a powerful WPF user interface architecture, multiple extensible plug-in architectures, and other original frameworks meant to support multiple products. Written in C#, some, but not all, of the technologies used extensively by the system were WPF (MVVM), WCF, MEF, AOP using PostSharp, COM, and Extended MAPI.
- Designed and created a new WPF user interface used by all desktop client products created with development platform mentioned above, offering a much more modern looking and acting user interface than what was offered by legacy versions of client products.
- Successfully planned and executed a unique strategy that involved reducing the level of maintenance work on a legacy product to a point where the company would have enough time and resources to introduce a completely new software product (produced by the development platform mentioned above), while still being able to take on new customers with the current product.
- Led the process of integrating client technology featuring a WPF user interface produced by the development platform mentioned above into the software ecosystem of company that acquired previous company with the intent that it would serve as a replacement for a highly visible client software product being offered at the time. Helped in part by the flexibility offered by the development platform, integration of the desired technology with the acquirer's back-end software was successfully achieved.
- Designed and developed a new calendaring web application using ASP.NET, Entity Framework, AJAX, and JSON to replace a legacy web front-end for a meeting scheduling system. This web application allowed users to easily create, view, and edit reservations within the booking system in a manner similar to how one interacts with an Outlook calender.
- Overhauled the complex publishing process of a legacy software product by designing and implementing a fully-automated build system which directly resulted in an immense amount of improved worker productivity and the ability to serve customers exponentially faster, greatly increasing general customer satisfaction.
- Led the integration of third party LCD screen technology with a meeting scheduling system product by designing an ASP.NET web connector that acted as a bridge between both technologies and then managing a team of developers in their implementation of said designs.

- Managed and participated in the design, implementation, and demonstration of new green technology produced by integrating a meeting scheduling system's ASP.NET web services with a widely used building control system in a coordinated effort with the building control system manufacturer's engineers. This allowed the scheduling system to directly affect building controls including AC, lights, and auxiliary hardware such that their use would shaped to closely fit the use of associated assets (e.g. rooms, conference centers).
- Designed a framework that resulted in several system services being created that were able to synchronize information from a customer's Exchange calendaring system with a meeting scheduling system product's back-end using C#, MAPI, and Exchange Web Services technologies.
- Led the process of integrating support for a major telecommunication provider's system into a meeting scheduling system product. This was achieved through the consumption of a new REST API provided by the telecommunication provider by the meeting scheduling system product's own ASP.NET services.
- Led the integration of support for BlackBerry and other mobile devices into a software product by designing an advanced and unique Microsoft Exchange synchronization solution utilizing Exchange Web Services (EWS) and Exchange Transport Agent technology and then managing a team of developers tasked in its implementation.
- Created new Microsoft Installer packages (MSI) using WiX for a number of software products, all of which were able to be compiled during an automated publishing process. Many of these packages were expected to be able to be deployed across the entirety of many large organizations without incident, and were able to do so at a near perfect rate.
- Added a powerful search interface for searching through work requests generated through a meeting scheduling system to an existing ASP.NET web application.
- Created a mail spooler transport system service that listened for and then reported on all transactions relating to a particular meeting or reserved asset made through both ASP.NET web application and desktop front-ends.
- Improved productivity and general quality of code for a number of different software teams through the establishment of accepted best practices as well as conventions meant to ensure consistency in both function and form of code produced by these teams.
- Developed a procedure for creating .NET assemblies that, when running, would generate detailed execution trace reports able to be displayed in chart form, allowing developers to see step-by-step execution of events that occurred during process lifetime. When used, this greatly assisted efforts to resolve issues that were difficult to reproduce in QA environments.
- Added robust support for the creation of Microsoft installer patches (MSP) into the automated build system, resulting in painless update experiences for customers, as well as mitigating risk for the customer due to the rollback capabilities of this patching technology. Few software companies are able to provide these kinds of patches due to both the complexity involved and the expertise in Windows Installer technologies required.
- Created a custom stress testing tool using C# that was used to simulate high traffic load on a meeting scheduling system (through both its client API and ASP.NET web application frontend). This allowed others developers to be able to discover and correct severe performance bottlenecks, vastly improving the performance of the system when under load.
- Created a COM shim library using C++ (ATL) that securely exposes .NET technology to Outlook by using the techniques of both containment and aggregation as well as acting as a CLR host. This allows for specialized handling of low level COM Interop concerns, as well as the ability to tailor the loading of the exposed technology so that its impact on Outlook's start-up times is minimal.
- Created a general-purpose Aspect Oriented Programming framework using C# and PostSharp that streamlines the way various crosscutting concerns are addressed across products such as logging, tracing, error handling. Applications in which the framework has been used exhibit increased code quality, stability and developer productivity.
- Created a general-purpose WPF user interface framework that streamlines the manner in which MVVM principles are implemented into WPF applications. Use of the framework has allowed rapid development of multiple WPF applications requiring advanced WPF functionality.

- Created a comprehensive Outlook-related framework that overhauls the entire Outlook Object Model and provides developers a safe and powerful way to integrate .NET technology with Outlook. Use of the framework allows developers the ability to focus on implementing both sophisticated and simple business concerns without having to worry about the many potential issues that can occur when interacting with Outlook or other Office products.
- Implemented a number of enhancements and changes into a claims tracking system used by various government agencies consisting of a VB.NET front-end and Oracle database back-end for the purpose of staying up to date with regulatory changes.
- Worked on bringing a number of ASP.NET web applications and C#/VB.NET desktop applications into compliance with the ADA. Among other things, this work included ensuring that said applications functioned well with JAWS screen readers.
- Created an ActiveX control using C++/MFC that collected information on relevant installed products (application names, versions, etc.) and then allowed for this information to be submitted to the software developer in order to aid the providing of support to the customer. The control featured a graphical directory dialog control that allowed the user to choose from a list of directories to perform the scan under.
- Maintained a number of legacy software products over the years, personally contributing thousands of improvements and bug fixes to these products and then eventually managing teams of people doing the same thing. These legacy software products, many of which had deep rooted issues, would end up becoming much more stable and better performing due to these efforts.
- Designed a packaging and deployment framework that offered the ability to provision automatic updates as well as a powerful Windows Installer bootstrapper written in C++ using the Win32 API. This bootstrapper even featured its own original scripting language for advanced deployment scenario handling.
- Maintained both client and server components for a legacy software product, fixing hundreds of bugs and increasing overhaul stability and customer satisfaction.

## Professional Experience

**IWCO Direct**, Chanhassen, Minnesota USA

*Software Engineer II* **Nov, 2020 - Present**

Working on improving software systems responsible for the processing of customer orders and coordinating the shipment of over 300 million pieces of mail per month.

**MTS Systems Corporation**, Eden Prairie, Minnesota USA

*Software Engineer* **Jun, 2019 - Oct, 2020**

Worked with development team responsible for creation of new automobile stress testing application suite.

**Starkey Hearing Technologies**, Eden Prairie, Minnesota USA

*Senior Software Developer* **Jul, 2017 - Apr, 2018**

Worked with multiple teams on user interface as well as back end of audiology software responsible for programming hearing aid devices.

**EMS Software**, Centennial, Colorado USA

*Lead Developer* **Jul, 2015 - Jan, 2017**

Lead designer, engineer, and manager of developer teams responsible for development of Outlook add-in client for meeting scheduling system.

**Emergingsoft**, Bloomington, Minnesota USA

*Senior Software Architect* **Jan, 2011 - Jul, 2015**

Lead designer, engineer, and manager of developer teams for meeting scheduling related products.

*Software Developer* **August 2008 - Dec. 2010**

Designer and engineer for meeting scheduling related products.

## Education

**University of Minnesota**, Minneapolis, Minnesota USA

B.Comp.E. (Bachelor of Computer Engineering) June 2008

## Expertise

- Design: Large on-premise and/or cloud-enabled n-tier architectures and designs to be used by thousands of concurrent users
- Languages: C#, C++, Assembly
- Frameworks and Technologies:.NET Framework, .NET Core, WPF, WCF, PostSharp, WiX, ASP.NET, MEF, COM, Active Directory and Group Policy
- APIs: Office, Extended MAPI, Win32, ATL, MSI
- Tools: Visual Studio 2019, WinDbg, Expression Blend
- Database Related: MSSQL, NHibernate
- Other: IL modification and analysis, performance optimization and analysis, static and dynamic disassembly, reverse engineering